P.O. BOX 177, BOX HILL, 3128. $2.00

MCCC INC NEWS

THE MONTHLY NEWSLETTER FROM THE FAMILY COMPUTER CLUB

A Chat with the President

Exposing BASIC Part 1 by Elaine Foster

The Editor's Bytes

C64 Op Codes

Software Spotlight - C64 Club Disk

Machine Language in Two Bytes by George Flanagan

Melbourne
Commodore
Computer
Club Inc.

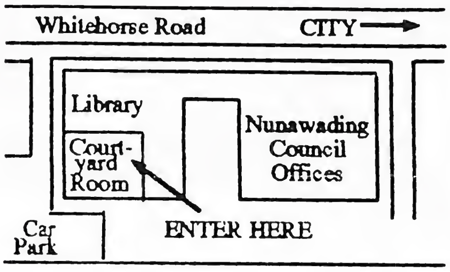
NOVEMBER 1994

VIC 20 C16 PLUS 4 C64 C128 AMIGA

Melbourne Commodore Computer Club Inc.

Postal Address:
P.O. Box 177,
Box Hill,Vic. 3128.

Club meetings are held on the second Wednesday of each month at the Nunawading Civic Centre in the Courtyard Room.



Meetings begin at 7.30 p.m. Please make an effort to arrive on time so the meeting can begin with no delays.

Contents of this newsletter may not be reproduced without the written permission of the MelbourneCommodore Computer Club Inc. Other clubs with whom we share reciprocal rights may reprint articles with acknowledgement.

All opinions expressed are those of the authors of individual articles and not necessarily those of the MCCC Inc.

Newsletter Editor:
Dorothy Millard
12 Venetian Court,
Croydon, Vic. 3136.
Telephone (03) 725-0682
Articles for the newsletter may be hardcopy (handwritten is equally acceptable), C64 or Amiga disk. All club members are invited to submit articles.

Secretary:
Robert Morrow
P.O. Box 651,
Templestowe, Vic. 3106.

DEADLINE FOR NEXT ISSUE
2nd December, 1994.

In this Issue.....

MEMBERSHIP FEES - Visitors Welcome
$35 per year Family Membership
$3 VISITOR (family - includes FREE Newsletter)

BENEFITS OF MEMBERSHIP

* Monthly newsletter mailed to all members.
* Pedlar's trading table selling recycled computer wares and blank 3 1/2" and 5 1/4" disks at competitive prices.
* Magazine Library - Magazines are available for members to borrow.
* Access to the club's extensive C64 & Amiga PD libraries - club disks available at meetings for only $2 each.
* Help is available to assist in solving problems.
* Computers for members' use.
* Monthly demonstrations

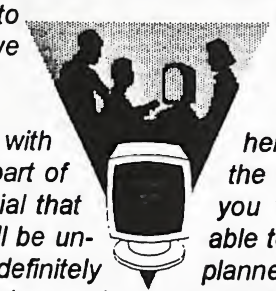
NOVEMBER DEMONSTRATIONS
Disks - How they work - Bernie
C64 - Reading the Stack Pointer by George
Amiga - PC Task by Grant

COMMITTEE FOR 1994/95 CLUB YEAR

PRESIDENT	Bernie O'Shea
SECRETARY	Bob Morrow
SECRETARY'S ASSISTANT	Jim Davies
TREASURER	George Flanagan
EDITOR	Dorothy Millard
C64/128 LIBRARIAN	ShirleyYoung
AMIGA LIBRARIANS	Brett Eden & Darryl Hunter
PEDLAR/DISK SALES	
MAGAZINE LIBRARIAN	Grant Davies
PUBLICITY	Brett Eden
COMMITTEE MEMBER	Rob Jackson
COMMITTEE MEMBER	Leoni Parsons

The Editor's Bytes

Wasn't it great to see so many new faces at the last meeting. It was unfortunate that the Amiga didn't make it to the meeting and we were unable to present the advertised demo on this machine. Our apologies for this. Steps have been taken by your committee to ensure that this doesn't occur again and next month we have something special to show you on the Amiga.



Didn't Elaine do a good job with her talk on programming on the C64? If you enjoyed this part of the evening please let us know, or if there is anything special that you would like to look at in the future. At this stage we will be unable to continue programming until the new year but more is definitely planned for the future. Our thanks to Elaine for all the work she put in. Amongst the pages of this newsletter you will find an article by Elaine about programming on the C64. Also on a club disk will be the programs she is working on.

The club thanks Adam Johnston who donated about 300 magazines, as well as some magazine disks, to the club library. Your donation is much appreciated. This means that our club magazine library now contains a good selection of Amiga magazines, as well as C64 ones, so if there is anything you are looking for pay Grant a visit and ask his assistance, or if you just wish to browse, the the magazines are there for you to borrow.

This month the Pedlar's Table should be worth visiting, with cheap magazines and various IBM bits and pieces, which should be worth a second look, as well as all the usual odds and ends. Blank disks are available as usual - note however that we will now be carrying High Density 3½" disks. If these sell well then we will be carrying them in future. Speaking of IBM's, a machine should be at the next meeting for those people who own one of these machines (and I know a number of people within the club do), or if you are just interested to see what it can do.

For C64 users a double sided catalogue disk is now available from the public domain librarian if you are interested in finding out what is in the club library and ordering back issues of the monthly club disks. Cost of the catalogue disk is only $2. How about the club running out of monthly disks at the last meeting? We certainly had a run on them, but if you missed out the librarian will have extra copies next time.

If you are an Amiga user looking for public domain software then Brett is the one to see. Lots of software is available, just let him know what you are looking for and he will be pleased to help. Most of the club's Amiga PD software is on compact disk, but easily copyable to floppy. If there is sufficient demand we will revert to preparing a monthly Amiga disk, so if this interests you please let us know.

Did any of you visit the Cyber Art Exhibition organized by The Rotary Club of The Basin (Inc). at the Bayswater Community Centre, as advertised in MCCC News? I did but regretted it as it had been cancelled. Nice of them to let us know wasn't it? That'll be the last time I publicize something for them.

With all the interest recently on programming, I thought the following tip might be useful: Beware the shifted space. A normal and (accidental?) shifted space will look exactly the same in a program listing, but will have different ASC values. This can cause an apparently perfect program to crash/fail. A VERY frustrating occurrence. The solution is to simply retype the line if in doubt.

Until next month..........

Dorothy.

$$$$$ PEDLAR'S CORNER $$$$$

Make yourself some money and sell your superseded computer wares at Pedlar's Corner. Bring goods to be sold along at around 7.00 pm. and pick up your money and anything left at 9.15 pm.

Please complete a form detailing the goods to be sold and the price wanted. Forms are available from Pedlar and are printed periodically in the newsletter. Please also ensure that goods are labelled with your name and the price required.

Note a commission of 10% is payable to the club.

At the next meeting there will be a selection of IBM books and bits and pieces on the Pedlar's table.

Magazines, Magazines and more Magazines

If you are interested in obtaining some of the classic Commodore C64 magazines, including Run, Compute, Compute's Gazette etc. then check out the Pedlar's Table at the next November club meeting.

FROM THE CLUB SHOP (alias Pedlar)

BLANK DISKS	5¼"	$6 for 10
	3½" Double Density	$9 for 10
	3½" High Density	$9 for 10

Please help by bringing the right money if possible

Welcome to the following new members:

Allan Swinden
Russell Alphey
Craig Stevens
Keith Chamberlin
Alan Share
Philip Seely
K. Sayers

Tea and coffee is available free of charge at each meeting in the kitchen, which is opposite the entrance.

Please help yourself and when finished wash, dry and put away your cup. Thank you.

A Chat with the President

This month's deadline has come around with a rush, and I find that I must write this and try to get it to Dorothy all within a few hours. I must admit that I have not had a very busy computing month in October, because there always seemed to be other duties calling which don't allow time to spend at the keyboard. Things like painting the living room, and dealing with the jungle outside. (And not too successfully either!!) The computer has had a little bit of routine use by my daughter doing some University assignments, and the only thing I have really done is assemble a disk of graphics and printer utilities for one of our more distant members who is having trouble getting labels with graphics to print on his unusual printer. If anyone is familiar with the AEG Olympia NP30 - apparently works as a MPS803 clone with extra capabilities - by all means give us a hoy! Of course I recommended "Fun Graphics Machine" as a program which caters for many different printers, and does labels, letterheads, business cards, as well as lots of other printing jobs. Never miss an opportunity to plug the work of a still-working and very clever programmer for the C64!

Over the last few months, my little effort for the newsletter (this page...you know Dorothy will let ANYONE write a page for the newsletter!) has been typing-free for the editor because it has been delivered by modem over the telephone. Dorothy uses the "Pagestream" desktop publishing program on her Amiga for arranging the newsletter, and I am writing this on a C128. The only way to avoid having to key all these words again is to somehow get them onto an Amiga 3.5" disk, and using a modem is the best way to achieve this. It also saves a stamp and about 24 hours. All I do is save this document to 5.25" disk as a Sequential file in "True" ASCII format - most word processors can do this - The Write Stuff certainly can - and transfer (upload) it to either Dorothy's or Brett's Amiga. It can then be read into any Amiga word processor or desktop publisher for editing or formatting. A real time saver. Last time I did this, Brett sent me a file he had recently downloaded from a bulletin board concerning the continuing saga of Commodore International, and I was then able to load it and print it out using my system. I guess this is possible using just about any computer, and I know Dorothy has transferred both text and graphics from Amiga to IBM. Without a modem, it helps if the disk size is the same.

Speaking about the dreaded "I" word, there is news from the committee that will interest those who have been talked into purchasing one of these machines. It's easy enough to understand if you have to use one at work or school, and you have to do some sort of homework. The transfer method described in the last paragraph is a bit cumbersome if you have to use it on EVERY file! The policy we worked out is that IBMs will not be banned from meetings. Although the club will not be purchasing any IBM equipment or software, if any member wishes to bring their own equipment to a meeting, it can be available for discussion or demonstration with other interested members. After all, one of the main attractions of our club is the ability to talk about computing with other knowledgeable members. We don't want familiar faces to just disappear when they buy a new computer. Quite often, they keep their old one too, and are still a valuable font of knowledge about their first loves.

We will not be devoting any time to formal demonstrations on the IBM platform as time at meetings is usually too short for what we have planned already. We will just have to see how members react to this change. As I said in my piece of two months ago, if the MCCC of the future consists of mainly IBM and/or Amiga owners, and club meetings have little or no C64/128 content, then I will be among the first to abandon ship and try to find a club which supports MY computer.

Bye for now..........Bernie.

PART 1

Exposing Basic in a C64

by
Elaine Foster

The following is background for talks given at the MCCC Meeting or to be given. If you need further background on anything, simply look at the C64 User Manual. It's all there and not hard to understand. The main reason for these talks is to get you interested in reading it, and to see that programming is not only not hard but it can be fun.

But if you want to read Shakespeare you have to learn English. If you want to programme you have to learn the language, BASIC, but it is much easier than learning English: only about 5 dozen words (60.000 for your decimalites!) If you learn them you can have lots of fun, astound your friends and fix crook
programmes, and lots more.

You will get a lot more out of all this if you will turn on your computer and actually type in the things I talk about here.

When I put programmes into upper case letters I don't mean for you to press the SHIFT key; when you type stuff in it comes out naturally in upper case on the screen.

If you wanted it to come out in lower case you would have to press the C= and SHIFT keys at the same time, but that is not what we are doing here.

I capitalise SHIFT, RETURN, etc. because that is the way they are-labelled on the keys.

1. IMMEDIATE VS PROGRAMME MODES

The Cursor - Move it around with the CRSR keys. Delete characters with the DEL key, make a space to insert characters with the INST key (viz: SHIFT+DEL).

Typing skills - (It takes you only 5 minutes to learn touch typing with all fingers, and a couple of months to become reasonably good at it. It's well worth it but if you are an individualist you can still programme with two fingers; it only takes longer.

Immediate (or Direct) Mode:
What you type onto the screen.

```
PRINT "HELLO" <RETURN>
```

Programme Mode:
Do the same, but put a line number ahead of the command:

```
10  PRINT "HELLO" <RETURN>
```

does not print HELLO yet, because it puts that line into a program.

2. KEYWORDS

These are just the language of BASIC. You will find them summarized on the back inside cover of the manual, described inside of it, eg:

```
RUN
LIST
PRINT
GOTO
FOR.....NEXT
REM
and so on
```

<RETURN> means press the RETURN key.

When you type RUN <RETURN> the programme you have just typed in prints HELLO. Easy? If you type LIST <RETURN> it lists what is inside the programme. PRINT of course prints, as long as you put in those double quote marks. GOTO100 goes to line 100. And so on. Is that complicated?

By the way you do not need to put in spaces, so don't; they just take up space, and in the C64 there isn't much of that. The only exception is what happens inside the quote marks. There, when you want a space to be printed you use a space.

3. A SIMPLE PROGRAMME

Type in the following line, pressing the <RETURN> key each time, of course:

```
10   PRINT "COME TO BASIC
     WEL "
```

The amount of things you can do

(Continued on page 7)

(Continued from page 6)

with this is amazing.

Experiments:

A. RUN that, viz, find a clear line, type RUN and then press <RETURN> key.

B. Add: (that is to say, simply type it and press <RETURN>):

```
20    GOTO 10
```

RUN that now. To stop the display press the RUN/STOP key. You can see that the stupid computer has done exactly what you told it to do; Print stuff and then print it again and again...

C. Add a semicolon (;) at the end of line 10, and of course press <RETURN>. RUN that. See what happens to the WEL? As before press RUN/STOP to stop.

Enter LIST (that is to say, type LIST and then press the <RETURN> key) to list that programme anytime you are in Immediate Mode. Enter RUN anytime you want to run it.

D. Introduce an error, eg. SPRINT instead of PRINT (use the INST key to make room for the S, ie. press SHIFT plus the DEL key). Press RETURN as usual with the cursor on that line, move the cursor to a clear area and enter RUN. Won't run will it? But it tells you why.

This is an Error Message, and there is quite a lot of them, listed on pages 159-160 of the Manual. Repair the damage (with the DEL key) and for line 20 type GOTO30, and see what happens when you RUN it. Another error.

E. The FOR...NEXT command. Add another line. type (and press <RETURN>):

```
15    FORT=1TO1000:NEXT
```

or if it really bothers you to leave out the spaces

```
15    FOR T=1 TO 1000: NEXT
```

RUN this. The delay is about 1.1 seconds for 1000 thingos. If you make that 10000 you get an 11 second delay. This is how you get a time delay in a programme. Obviously that can be useful.

The FOR...NEXT command can be very useful for something else: It saves a lot of typing. Prove this to yourself. Delete the semicolon (Use DEL key carefully and again press <RETURN>). Type in a new line 15 (automatically overwrites the previous one) and press <RETURN>:

```
15    PRINT "COME TO BASIC
      WEL"
```

and add a REM to line 20 (USE INST to make a space for it):

```
20    REM GOTO10
```

If you again LIST the programme now you get:

```
10    PRINT"COME TO BASIC
      WEL"
15    PRINT"COME TO BASIC
      WEL"
20    REM GOTO10
```

And if you RUN it you get two lines printed; the REM causes GOTO10 to be ignored when you run the programme. If you repeated the PRINT command 10 times or 100 times it would print 10 or 100 lines. But it would be much easier to use FOR...NEXT:

```
5     FORN=1TO2
10    PRINT"COME TO BASIC
      WEL"
15    NEXT
20    REM GOTO10
```

RUN this and you get the same two lines. Change the 2 to a 5 and you get five lines. Easy? Now put the semicolon in at the end of line 10 and run again. The semicolon obviously runs PRINT commands together. Very useful sometimes.

F. Using the Colon (:)

Type NEW <RETURN>. Now LIST. What do you get? What happened to the programme? It is gone (it is still there, but invisible unless you know how to recover it.) Now type in this new programme:

```
10    FORN=1TO2:PRINT"COME
      TO BASIC WEL";:NEXT:
      GOTO10
```

RUN that. Same result. The colon gives the same result as a new line, and saves a lot of space. But you can only put two screen lines into one programme line. If you type in more than that you lose it.

Modify that long line 10 slightly:

```
10    FORN=1TO2:PRINT"COME
      TO BASIC WEL";:NEXT:
      PRINT:GOTO10
```

RUN that and you see that the extra PRINT command has broken up the lines which were run together by a semicolon. This is useful when you want to run words together up to a point, and then not.

4. VARIABLES

In Direct Mode, enter the following statements, pressing

(Continued on page 8)

(Continued from page 7)

<RETURN> each time, of course, and watch what what happens:

```
X=4 (X is a “variable”)
PRINTX
PRINTX+7
V=5
PRINTX+Y
PRINT2
PRINTLOG(2) (this gives
the natural logarithm of 2)
PRINTLOG(2)/2.303 (this
gives log 2 to the base 10)
PRINT10
PRINT"10"
PRINT-10
```

The last three show you something. PRINT10 puts a space before the 10. PRINT"10" does not; it just prints what is inside the quotes. PRINT-10 puts a negative sign in front of 10, so that is what the space was for: the space alone means it is a positive number. When the 10 is in quotes, it is called a “string.” When it is not, it is called a “variable.”

5. STRINGS

Variables that are words, not numbers. They are always put inside quotes: In Direct Mode enter the following and watch what happens:

```
A$="PIECE"
PRINTA$
PRINTA$+"OF CAKE"
B$="OF CAKE"
PRINTA$+B$
A$="10"
PRINTVAL(A$) (this gives
the numerical value of the
number inside the quotes)
PRINTVAL(10)
```

What happens in the latter case? What does this tell you? Try PRINTVAL(“10”).

Incidentally you are supposed to call A$ “A String”, but like most people I just call it “A Dollar.”

6. CURSOR POSITION

A. Enter NEW in Direct Mode and then enter the following programme. [down] means press the cursor down key:

```
10   PRINT" ONE"
20   PRINT" TWO"
30   PRINT" [down]"
40   PRINT" THREE"
```

RUN this and watch what happens.

Replace the cursor command in line 30 with each of the following, and see what happens when you RUN it: [up], [right], [left], [clr], [home], [ctrl 9] (this turns reverse video on), [ctrl 2] (for white, or whatever key you want for a different cursor colour).

B. Look at the odd characters you get on the screen when cursor or colour characters appear within quotes. They are weird and when you list the programme make it very hard to tell what has been pressed.

If you have a printer, you can make a table that tells you what each strange character means. Enter NEW, and then enter the following programme:

```
10   OPEN1,4:PRINT#1:CMD1:
     REM THIS MAKES IT
     POSSIBLE TO PRINT
     YOUR LIST
20   PRINT"[left] = CRSR LEFT"
30   PRINT"[right] = CRSR
     RIGHT"
40   [On each line substitute
     whatever you want deci
     phered from the above list.
     Use a separate line number
     for each line, so the next
one  would be 50, the next is 60
     and so on.]
1000 LIST
```

Now in direct mode enter:

```
PRINT#1:CLOSE1
```

This tidies everything up. Note that LIST can be used from within a programme but it then ends that programme. No lines can be added after it. That is why PRINT#1:CLOSE1 had to be entered manually.

If you do not have a printer omit line 10 and the PRINT#1;CLOSE1 afterwards, then enter this programme anyhow and it will give you a nice screen display.

I have *no* idea why Commodore used those awful reverse video things to show cursor commands.

If you have an XETEC or other suitable printer interface it translates them to ordinary words, as above, making life much simpler.

7. A DEMONSTRATION PROGRAMME WITH SIMPLE GRAPHICS.

In this programme, SHIFT/U means hold down the shift key and press the letter U at the same time, and so on for other keys. [9 rt] means press the CRSR-right key 9 times. [17 spc] means that you press the space-bar 17 times. And so on.

```
10   POKE53280,0:POKE53281,
     0:POKE646,1:REM
     BORDER SCREEN,
```

(Continued on page 9)

(Continued from page 8)

```
     CURSOR COLOURS
20   PRINT"[clr]":REM CLEARS
     THE SCREEN
30   PRINT [6 dn]:REM MOVES
     CURSOR 6 LINES DOWN
     FROM HOME
40   PRINT"[9 rt][SHIFT/U]
     [17 SHIFT/*][SHIFT/I]:REM
     TOP OF GRAPHIC
```

RUN it now, just to see what you get.

This business of running a partly finished programme can be useful in "debugging", to show you possible errors.

You can get the same result in the middle of an already completed programme by introducing a :STOP: somewhere.

```
50   PRINT"[9 rt][S/B][17 spc]
     [SHIFT/B]":NEXT:REM 2
     SIDES
```

If you have understood what I have done so far, you will realize that if I used FORN=1TO7 in line 60, I could eliminate line 50 entirely. Do this now: find a clear line, type 50 <RETURN> and line 50 is gone. Now using the same technique as for line 60, change line 40 to line 70 and press <RETURN>. LISTing it again will show that line 40 has been repeated, now as line 70.

Change it to this:

```
70   PRINT"[9 rt][SHIFT/J]
     [17
SHIFT/*][SHIFT/K]":REM
     BOTTOM OF GRAPHIC
```

Continue with:

```
80   X=68:Y=5:A$="X =":
     B$="Y ="
```

Now we print these variables inside the borders of the box we have produced:

```
90   PRINT"[7 up][CTRL/8][11 rt]"
     A$X:REM PRINTS INSIDE
     THE BOX
100  PRINT:REM BLANK LINE
110  PRINT"[CTRL/9][11 rt][spc]"
     B$Y
120  PRINT"[HOME][CTRL/2]":
     REM CURSOR TO HOME
     AND WHITE
```

Carefully check this on the screen with the programme on paper, and then enter SAVE "LESSON",8 with a formatted disk in the disk drive.

When you have saved it, *then* run it. If you have made some catastrophic mistake (let's say it locks up the computer) at least you will be able to turn everything off and reload.

To LOAD it, enter LOAD "LESSON",8 and this disk programme will be loaded into memory. Again, LOAD loads the programme from disk, SAVE saves it to disk. Who said programming was complicated?

When you run that programme, you ought to see a nice white box with yellow numbers inside and a white cursor at "home" position. Insert STOP at various places in he programme to see just what is happening, eg.

```
100  PRINT:STOP:REM BLANK
     LINE
```

will stop the programme before it prints the numbers, and you should get only a yellow cursor inside the box. Interrupt it at other places (but be sure to omit the STOP in prior lines.)

When you have interrupted it, move the cursor to a clear space and enter CONT and lo the programme will continue. If you get a "CAN'T CONTINUE" error it means you have made some change. CONT only works if you have changed nothing. See, you have learned one more keyword.

You can have fun with this by changing the graphics in lines 40 - 70. The shapes of the various graphics are shown on the front of each of the keys on the keyboard.

8. TABULATION

Type in

```
10
     PRINT'HELLO","GOODBYE
"
```

and observe what happens when you run this. Try using two commas. Change to:

```
10   PRINT "HELLO"TAB(20)
     "GOODBYE"
```

and you can see that using a comma was the same as if you had used TAB(10). The number obviously refers to the number of screen spaces tabulated, just as on a typewriter.

When I can get back to this keyboard I shall continue these lessons. Please let Dorothy know whether or not these are useful..........Elaine

Software Spotlight

C64 November Club Disk
A review by Dorothy Millard

JUGGLER or Bubbles! This an animated graphics demo - doesn't do much but is fun to watch.

AMORTIZATION - This is one of the old Commodore Educational Software titles. It will amortize a loan based on information supplied by the user. It does the job and is simple to use.

PRINT SORT - Another old Commodore Educational Software title. This one simply alphabetizes a list of names. It is effective and simple to use.

ELIZA - The opening screen states "I am Eliza your Computer Psychiatrist, so please be honest with me....." You must answer a number of questions and the computer will "answer you back." Responses are a little slow. Not to be taken seriously the program is a bit of fun.

DISK UTILITY - This utility shows it's age by including facilities to copy the DOS Wedge programs which are rarely used these days, most people preferring to use a cartridge such as the Fast Load or Action Replay. Disk Utility supports the following:

1. Disk Directory
2. Format New Disk
3. Initialize Disk
4. Copy file on same disk.
5. Copy both DOS Wedge programs
6. Rename file
7. Validate Disks
8. Write Disk Utility
9. Error Status
10. Merge Files
11. Rename Disks and IDs
12. Scratch and Unscratch Files
13. Exit Program

Despite it's age Disk Utility is useful, especially the Merge Files and Unscratch Command. Remember when using Unscratch that it won't work if anything else has been written to the disk since the scratch was made.

DON'T WORRY BE HAPPY - This is a fun graphics and sound demo.

KILLER DRIVE II - I hope your 1541 disk drive doesn't "eat" disks like this one!!!!!

HIGH-RISE - A simple platform game - what more can I say?

COPS & ROBBERS - This program written by Mike Davis is a simple but fun game.

VCR INDEX - A typical public domain specialist database which is somewhat limited in it's application.

SPIKE - Spike is a classic public domain game which has been around for many years. It is played on a grid and you must enclose parts of the grid while avoiding the "spikes." It is a very simple concept but works well and is addictive.

GAP REMOVER - There are no instructions but I assume it removes gaps from disk directories.

RAINBOW TOWERS - This is a Towers of Hanoi puzzle in which you must transfer the disks from No.1 needle to No.3 needle one at a time, never putting a larger disk on a smaller one. It is a primative but workable rendition.

MUSIC - Synthesised samples of a selection of tunes. A pleasant interlude and a chance to hear music played on the C64.

EVIL RIDGE - The Kingdom of Picar is overrun by evil. Guess what? The only hope is for you to travel to the highest mountain in the kingdom - Evil Ridge. Your quest is to find the Mythical Sceptre of Lawfulness and only the Sceptre can save the Kingdom from the evil. This is a very basic adventure with a limited parser. A simple easy game with only twelve locations to explore.

HINT - (K)ick the snake - sometimes he will win and you will then have no choice but to start again - just keep trying until you succeed.

The above disk is available at the club meeting from the C64 Librarian for only $2.

October C64 Club Disk

The C64 Librarian apologises for running out of disks at the last meeting. They will be available next month. Sorry if you missed out.

Catalogue Disk

A double sided catalogue disk of the club's C64 public domain software will be available at the next meeting for only $2.

Dealer Directory

McGILLS AUTHORISED NEWSAGENCY 187 Elizabeth St., Melbourne (602-5566) COMPUTER BOOKS DISCOUNT 10%	**MEGATRON COMPUTER INDUSTRIES** 62 Charter St., Ringwood (870-4844) *All Commodore Computers Serviced* 10% DISCOUNT ON LABOUR CHARGE	**MAXWELL OFFICE EQUIPMENT** 162-164 Nicholson St., Abbotsford (419-6811) 10% DISCOUNT ON SOFTWARE
NOVO COMPUTER HARDWARE P.O.Box 237, Mayfield, NSW, 2304. (049-201102) MEMBER DISCOUNT		**M.V.B. COMPUTER SUPPLIES** 586 Dorset Rd., Croydon (725-6255) GOOD PRICES AND FRIENDLY ADVICE
COLLINS BOOKSELLERS 401 Swanston St., Melbourne 654-3144 COMPUTER BOOKS ONLY DISCOUNT 10%	**TECHNICAL BOOK AND MAGAZINE CO.** 295 Swanston St., Melbourne 663-3951 *Computer Books (not Magazines) and Selected Software* DISCOUNT 10%	**PRESTON ELECTRONIC COMPONENTS** 217 High St., Preston 484-0191 10% DISCOUNT

To obtain the stated discount you should produce your current club membership card.

C64 Op Codes in Decimal

CODE	Flags	im	z	z,X	ab	ab,X	ab,Y	i,X	i,Y
ADC	n,z,c,v	105	101	117	109	125	121	97	113
SBC	n,z,c,v	233	229	245	237	253	249	225	241
AND	n,z	41	37	53	45	61	57	33	49
CMP	n,z,c	201	197	213	205	221	217	193	209
EOR	n,z	73	69	85	77	93	89	65	81
ORA	n,z	9	5	21	13	29	25	1	17
LDA	n,z	169	165	181	173	189	185	161	177
STA	-	-	133	149	141	157	153	129	145

		im	z		ab	ab,X	ab,Y	Compare Tests
								CMP for A reg
LDX	n,z	162	166	z,Y 182	174	-	190	CPX for X reg
LDY	n,z	160	164	z,X 180	172	188	-	CPY for Y reg
CPX	n,z,c	224	228	-	236			
CPY	n,z,c	192	196	-	204	if reg < memory: c clear		
STX	-	-	134	z,Y 150	142	if reg >= memory: c set		
STY	-	-	132	z,X 148	140	if reg = memory: z set		

		accum.	z	z,X	ab	ab,X		
ASL	n,z,c	10	6	22	14	30	c←(b7)	(b0)←0
LSR	n,z,c	74	70	86	78	94	0→(b7)	(b0)→c
ROL	n,z,c	42	38	54	46	62	c←(b7)	(b0)←c
ROR	n,z,c	106	102	118	110	126	c→(b7)	(b0)→c
DEC	n,z	-	198	214	206	222		
INC	n,z	-	230	246	238	254		
BIT	(b7)→n (b6)→v		36	-	44	b7 & b6 are Memory bits.		

If M and A both 0 then 0→z. 36 also used to hop 1 byte. 44 hops 2 bytes.

CLI	0→i	88	CLC	0→c	24	CLD	0→d	216	CLV	0→v	184
SEI	1→i	120	SEC	1→c	56	SED	1→d	248	NOP	-	234
DEX	n,z	202	DEY	n,z	136	INX	n,z	232	INY	n,z	200
JMPabs	-	76	JMPind	-	108	JSR	-	32	BRK	-	0
PHA	-	72	PHP	-	8	PLA	n,z	104	PLP	stack	40
RTI	stack	64	RTS	-	96	TSX	n,z	186	TXS	n,z	154
TAX	n,z	170	TAY	n,z	168	TXA	n,z	138	TYA	n,z	152

BCC	144	BCS	176	BEQ	240	BNE	208
BMI	48	BPL	16	BVC	80	BVS	112

"stack" means the P (flag) register is loaded from the stack.
"Flags" are the flags that may be changed by the operation.
"i,X" addressing needs a string of base page addresses. Not there in BASIC!
SEI and CLI set and clear the interrupt DISABLE flag, respectively.

Machine Language in Two Bytes

by George Flanagan

In my BASIC VARIABLES series in 1993, I demonstrated, in Pt.2 "Garbage Collection" and "Out of Memory," errors caused by using up all BASIC program memory. However, a more likely cause of "Out of Memory" error for the novice programmer is overloading the stack.

The stack is the second page of memory and it is addressed by the microprocessor using an internal register called the stack pointer. There is only one way of reading the stack pointer and that is to use the TSX op code in a machine language program. This op code transfers the stack pointer to the X register.

SYS

You are probably aware that the BASIC command SYS jumps to the memory address that follows it, but before it jumps, BASIC loads the A, X, Y and P registers with the numbers in locations 780 through 783, respectively. For this reason 780 is called .A, 781 .X, 782 .Y and 783 .P. Immediately before returning to BASIC from a SYS command, the current values of the A, X, Y and P registers are stored in locations 780-783. In order to demonstrate the stack pointer and how bad programming can overload the stack, enter the following program:-

```
10  REM STACK OVERLOADING
20  POKE 1020,186: REM TSX
30  POKE 1021,96: REM RTS
40  SYS 1020
50  PRINT PEEK (781): REM .X
60  GOSUB 100
70  END
100 N=N+1
110 GOTO 40
120 RETURN: REM (NOT DONE)
```

The BASIC program stores 186 at location 1020 and 96 at location 1021. The 186 is the TSX op code while 96 is the "Return from Subroutine" (RTS) op code. This 2 byte machine language program is run by the SYS 1020 command and the value of X is obtained by PEEK (781).

When you run the program you will get a string of decreasing numbers, then an out of memory error. Why? Because the program never gets to execute the RETURN statement. If you print N from the keyboard you will learn how many times you can jump out of a subroutine before disaster strikes! In direct mode (try GOTO 40) you will also find that BASIC has cleared the stack pointer as part of its error recovery sequence.

The stack is used as a storage area managed by the microprocessor (65l0) on a last in, first out basis. The stack pointer indicates the next address for a PHA (PusH A) or PHP (PusH P) operation. When a GOSUB is executed, information required for the correct operation of the RETURN command is pushed onto the stack and used. The op codes for reading from the stack are PLA (for pull A) and PLP (for pull P). As the "push" codes decrement the stack pointer and the "pull" codes increment it, you will see why the stack pointer kept decreasing when GOSUBs were executed without an equal number of RETURN commands.

The FOR NEXT command uses many more stack entries (18) and you should not jump out of a FOR NEXT loop either. However my attempts to force an out of memory error from a FOR NEXT loop failed (BASIC was too smart for me!) Not to be denied, I added the following lines to the program above:

```
55  FOR J=1TO2
65  NEXT: REM (NOT DONE)
```

The double whammy really works fast! I hope you enjoyed that and learned at the same time - as I do when writing this stuff. Don't be overawed by the op code table - it's like a dictionary - you don't have to know all the words in it, in fact you would never use many of them at all.

Melbourne Commodore Computer Club Inc.

If undeliverable return to:

P.O. Box 177,
BOX HILL, VIC. 3128.

AUSTRALIA 45c
KANGAROO

Mr. R.J. ALPHEY
439 PUNT ROAD
RICHMOND, Vic. 3121.

C LUB MEETING DATES FOR 1994/95

12th January, 1994. 9th February, 1994. 9th March, 1994.
13th April, 1994. 11th May, 1994. 8th June, 1994.
13th July, 1994. 10th August, 1994. 14th September, 1994.
12th October, 1994. 9th November, 1994. 14th December, 1994.
11th January, 1995 8th February, 1995. 8th March, 1995.

Please Note:
All club meetings are on the second Wednesday of each month in the Courtyard Room, Nunawading Civic Centre, Whitehorse Road, Nunawading. The meeting room is available between 7-11 p.m.